

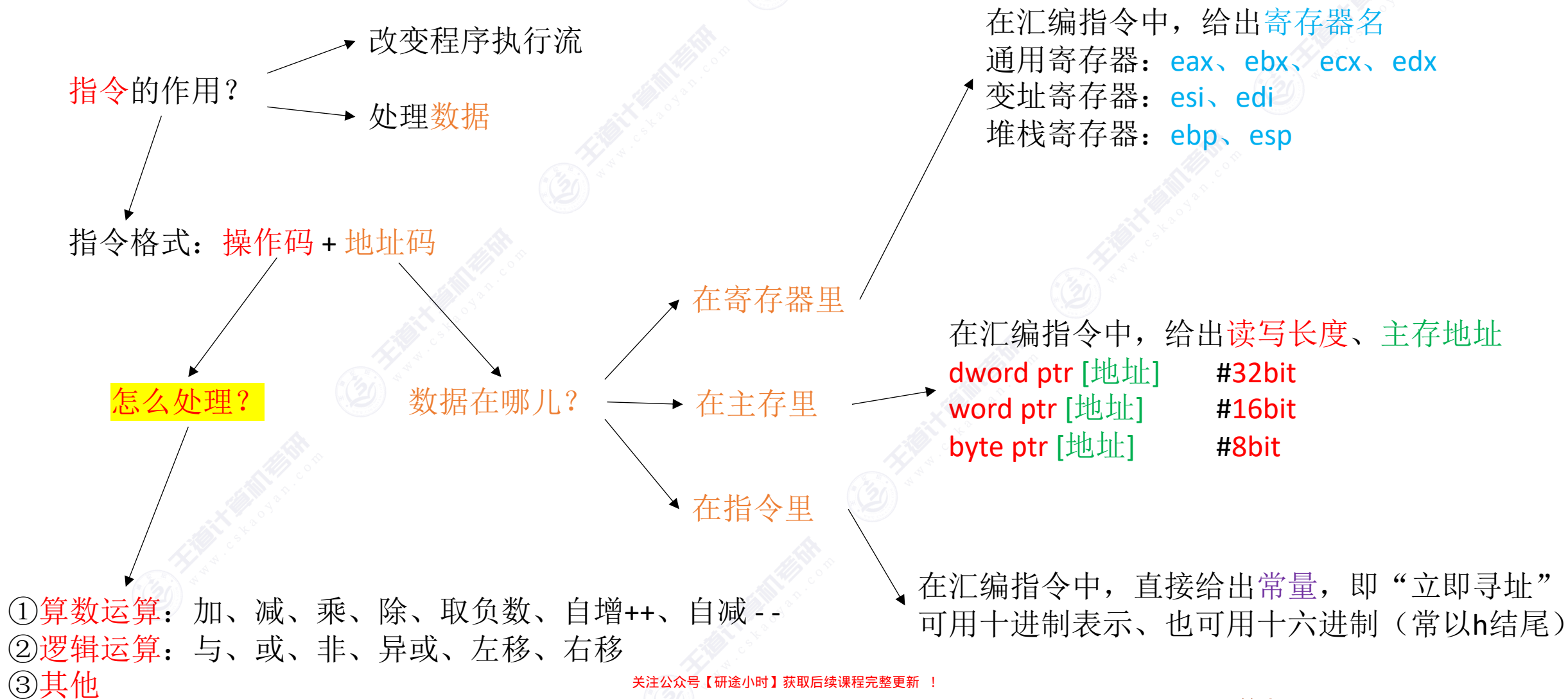
本节内容

常用的 x86汇编指令

关注公众号【研途小时】获取后续课程完整更新！

王道考研/CSKAOYAN.COM

本节总览



常见的算数运算指令

destination: 目的地 (**d** 目的操作数)
source: 来源地 (**s** 源操作数)

目的操作数 **d** 不可以是常量

功能	英文	汇编指令	注释
加	add	add d,s	#计算d+s, 结果存入d
减	subtract	sub d,s	#计算d-s, 结果存入d
乘	multiply	mul d,s imul d,s	#无符号数d*s, 乘积存入d #有符号数d*s, 乘积存入d
除	divide	div s idiv s	#无符号数除法 edx:eax/s, 商存入eax, 余数存入edx #有符号数除法 edx:eax/s, 商存入eax, 余数存入edx
取负数	negative	neg d	#将d取负数, 结果存入d
自增++	increase	inc d	#将d++, 结果存入d
自减--	decrease	dec d	#将d--, 结果存入d

关于王道书的解释

1) **add/sub 指令**。add 指令将两个操作数相加，相加的结果保存到第一个操作数中。sub 指令用于两个操作数相减，相减的结果保存到第一个操作数中。

它们的语法如下：

```
add <reg>,<reg> / sub <reg>,<reg>
add <reg>,<mem> / sub <reg>,<mem>
add <mem>,<reg> / sub <mem>,<reg>
add <reg>,<con> / sub <reg>,<con>
add <mem>,<con> / sub <mem>,<con>
```

举例：

```
sub eax, 10          #eax ← eax-10
add byte ptr [var], 10  #10 与 var 值指示的内存地址的一字节值相加，并将结果
                        保存在 var 值指示的内存地址的字节中
```



没有.....你听我解释

<reg>

寄存器



n. register

<mem>

内存



n. memory

<con>

常数



n. constant

常见的逻辑运算指令

功能	英文	汇编指令	注释
与	and	and d,s	#将 d、s 逐位相与，结果放回d
或	or	or d,s	#将 d、s 逐位相或，结果放回d
非	not	not d	#将 d 逐位取反，结果放回d
异或	exclusive or	xor d,s	#将 d、s 逐位异或，结果放回d
左移	shift left	shl d,s	#将d逻辑左移s位，结果放回d（通常s是常量）
右移	shift right	shr d,s	#将d逻辑右移s位，结果放回d（通常s是常量）

其他指令

用于实现分支结构、循环结构的指令：cmp、test、jmp、jxxx

用于实现函数调用的指令：push、pop、call、ret

用于实现数据转移的指令：mov



世上无难事，只要肯放弃



公众号：王道在线



b站：王道计算机教育



抖音：王道计算机考研